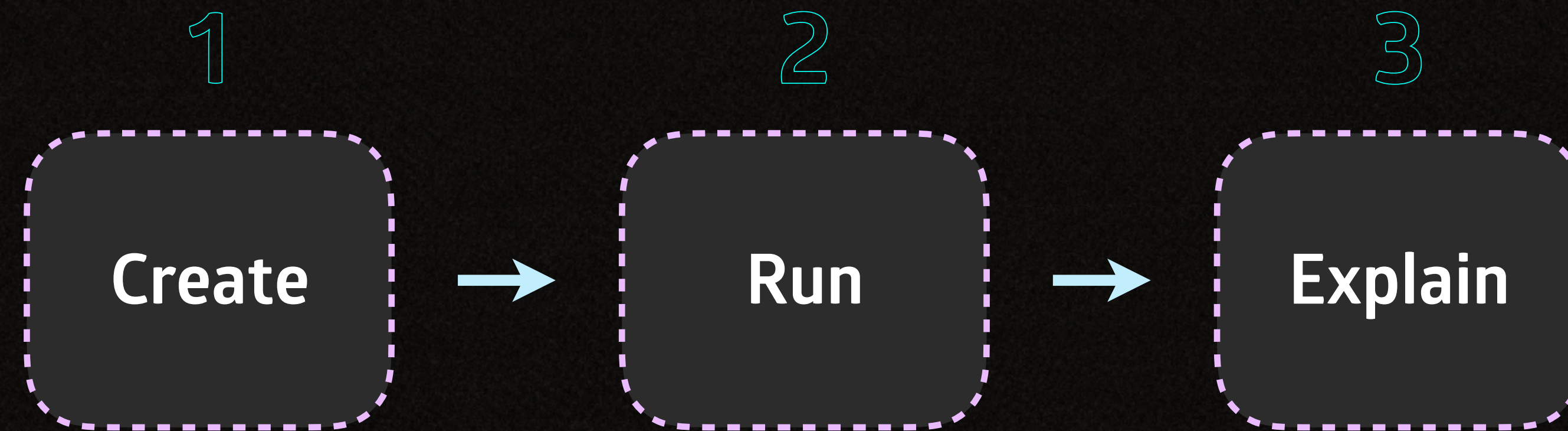


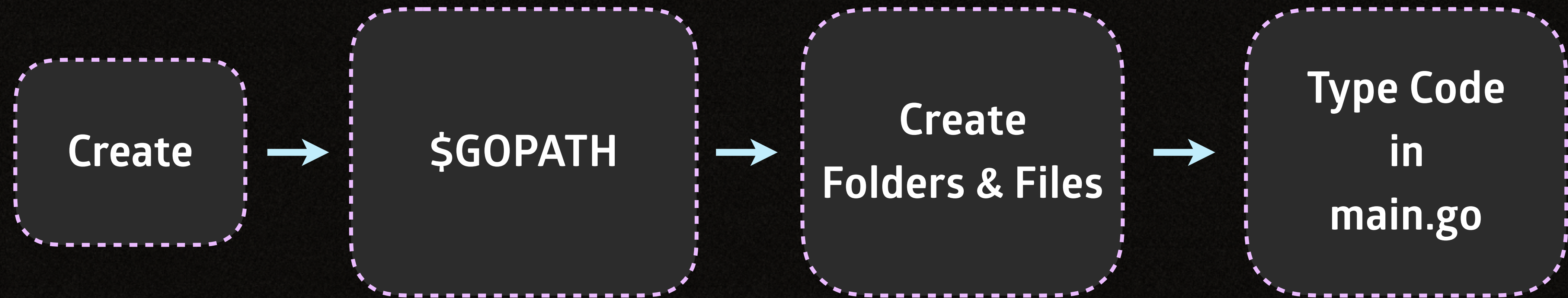
FIRST PROGRAM

You're going to learn how to *create* and *run* your *first Go program*



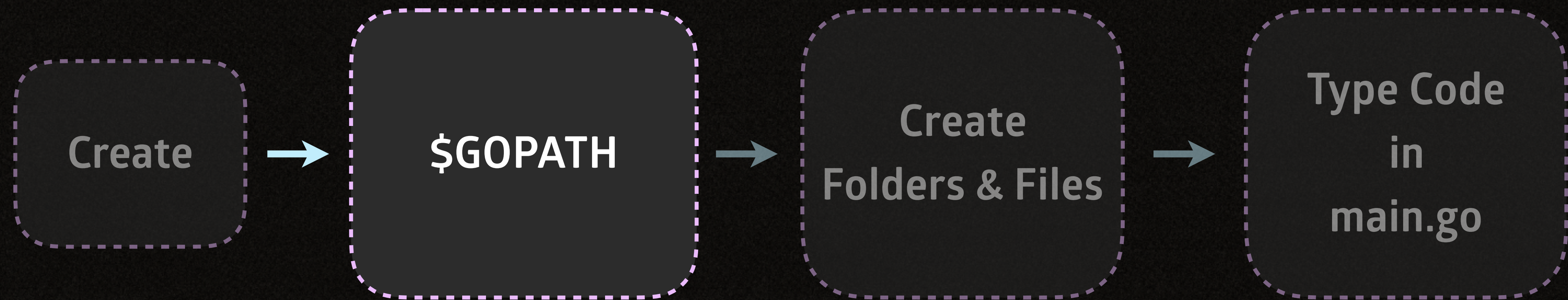
CREATE

*In this first step, you're going to **create** your first Go program.*



LEARN ABOUT \$GOPATH

You're going to *create folders* for your program



\$GOPATH

demonstration

\$GOPATH

an environment variable

by default Go assumes that it's in your **Users folder**

do not set your *\$GOPATH* variable

\$GOPATH

workspace

Go **tools** find Go **source-code** files

by looking at \$GOPATH

\$GOPATH

physical folder



automatically **creates** a **physical folder** for `$GOPATH`

*under your **users** folder
(its default location)*

\$GOPATH

src folder

contains source-code files

\$GOPATH

src folder

Go source code file



\$GOPATH

└─> /src

└─> /first

└─> /main.go

*an executable go program
needs its own directory*

\$GOPATH

src folder

Go source code file



\$GOPATH

➔ /src

➔ /github.com

➔ /youruser

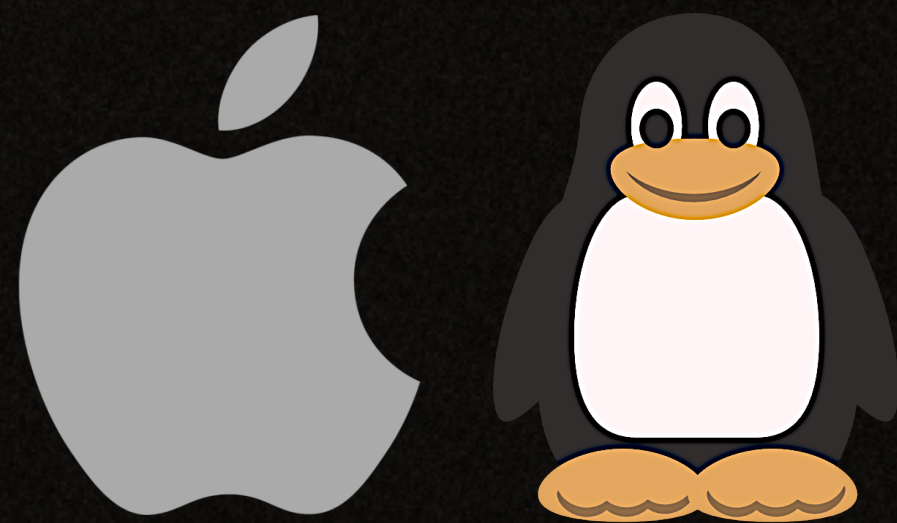
➔ /first

➔ /main.go

GOPATH

Where is your `$GOPATH`? Let's find it!

`$GOPATH`



`~/go`

`/Users/YourUsername/go`



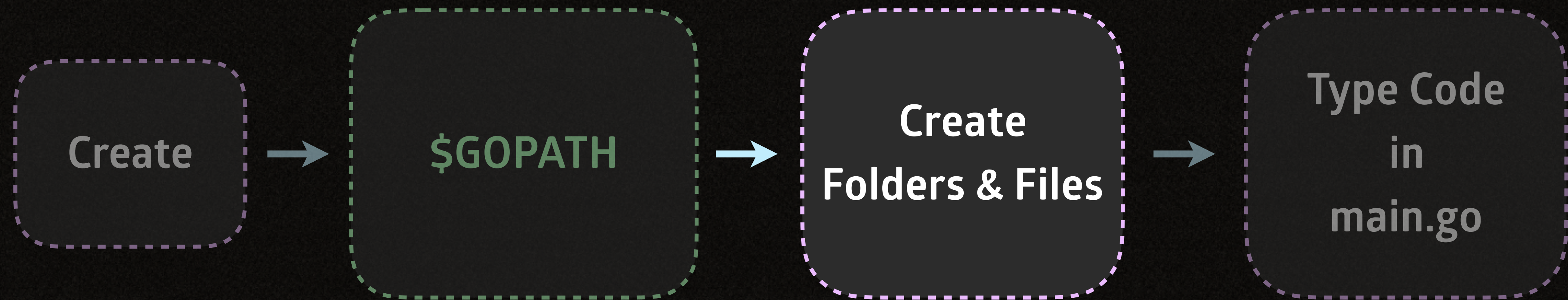
`%userprofile%\go`

`c:\Users\YourUsername\go`

`/c/Users/YourUsername/go`

CREATE FOLDERS & FILES

You're going to create *folders* and *main.go*



CREATE FOLDERS

Create these *folders* under your *\$GOPATH/src*

1 **learngo**

2 **first**

If you've a **github** account:

\$GOPATH/src/github.com/username/learngo/first

CREATE MAIN.GO

Create your first program file: "main.go"

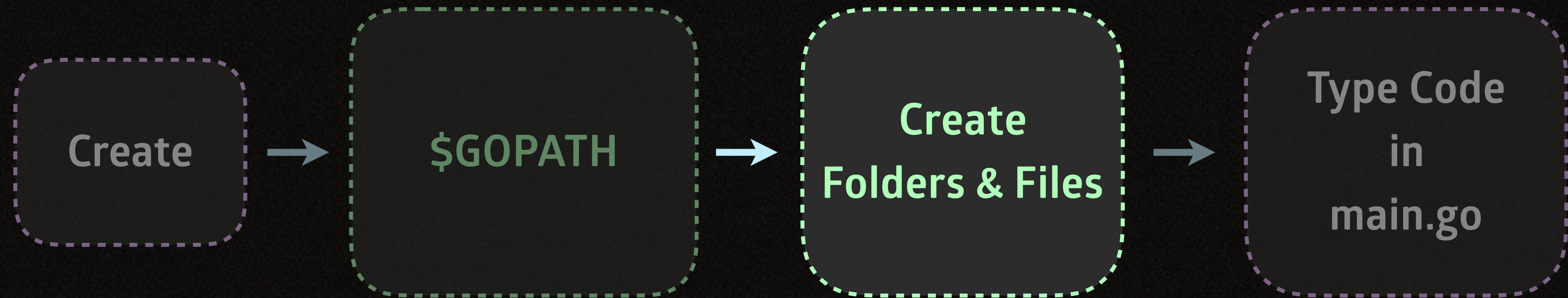


under:

`$GOPATH/src/github.com/username/learngo/first`

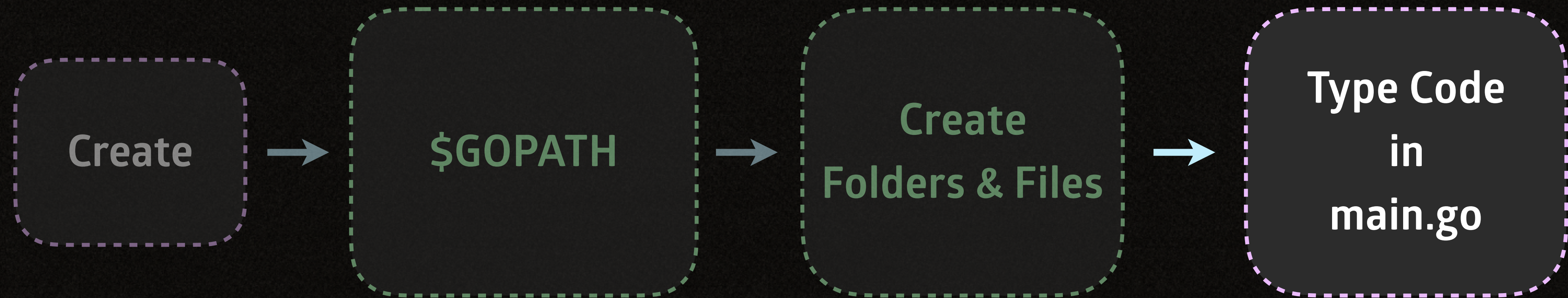
CREATE MAIN.GO

*You've created your first Go file, **congrats!***



CODE

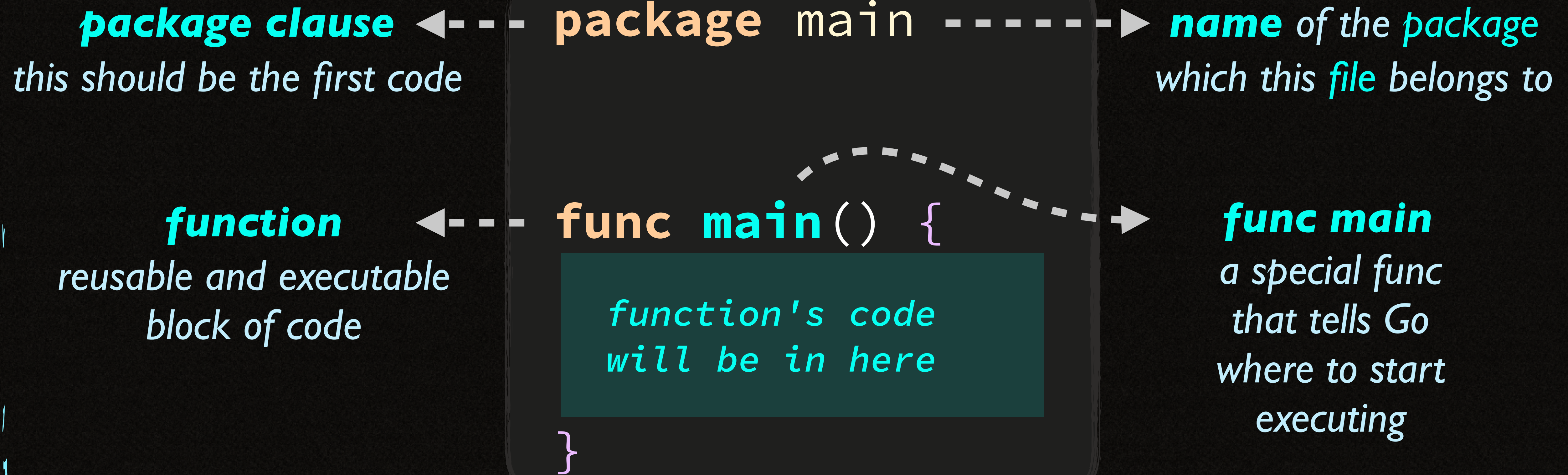
You're going to *code* your *first Go program*



CODE

Let's *code* your first Go program

main.go



CODE

I've removed the annotations from the code for clarity

main.go

```
package main
```

```
func main() {  
}
```

CODE

Let's *print something* to the console!

```
package main
import "fmt"

func main() {
    fmt.Println("Hello Gopher!")
}
```

HOW TO CODE WITH ME

*I'll always be providing the **full source-code** for the examples*

*sometimes,
i won't show these
common statements
in here*



```
package main
import "fmt"

func main() {
}
```

*full source-code
will always be
available on
github
&
playground*

github

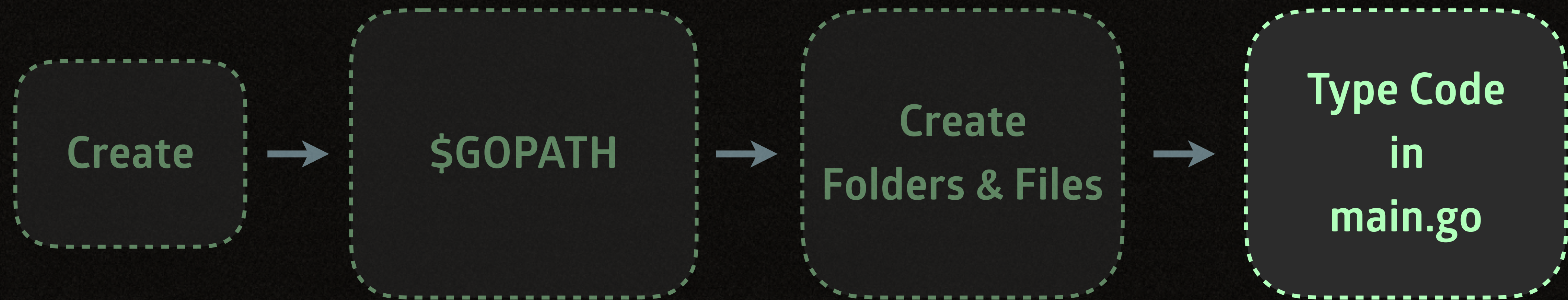
<https://github.com/inancgumus/learngo>

playground

<https://play.golang.org/>

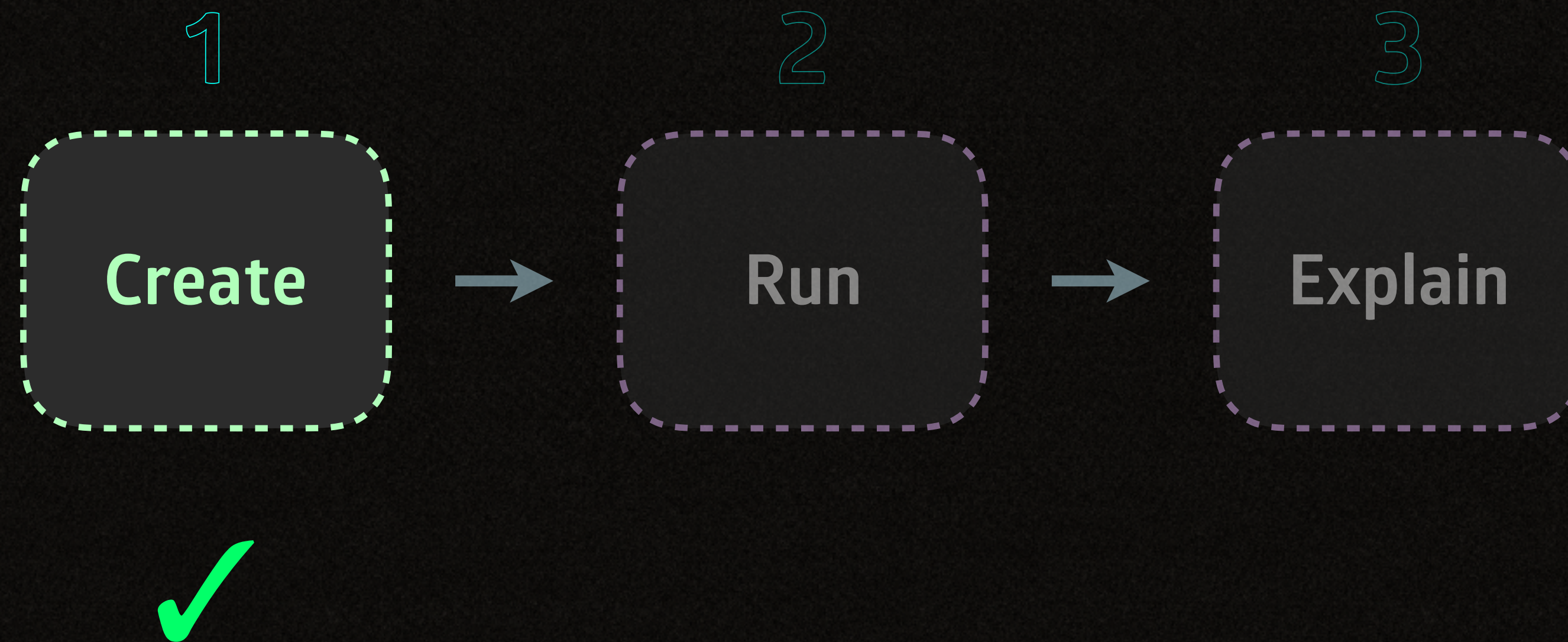
CREATE

You've *completed* creating your first program, *well done!*



FIRST PROGRAM

*Congrats! You've learned how to **create** your first Go program.*



RUN

You're going to *run* your first Go program.

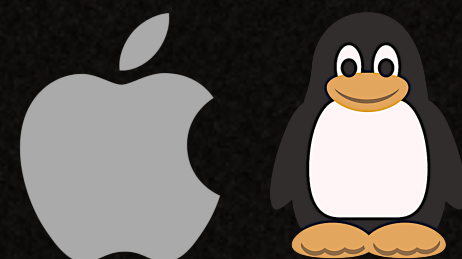


GO BUILD

You're going to learn how to *compile* your Go program using `go build` tool.



go build



./first



first

COMPILE-TIME

VS

RUNTIME

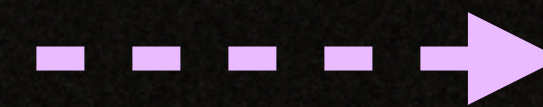
SOURCE CODE

```
package main
import "fmt"

func main() {
    fmt.Println("hello!")
}
```

Source-Code is like a seed
(compiled or not)

`$ go build`



Compiling

COMPILED CODE

```
MOVQ GS:0x8a0, CX
CMPQ 0x10(CX), SP
JBE 0x108e7c7
SUBQ $0x48, SP
MOVQ BP, 0x40(SP)
...
```



RUNTIME

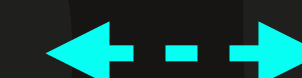
memory



`$ ./first`

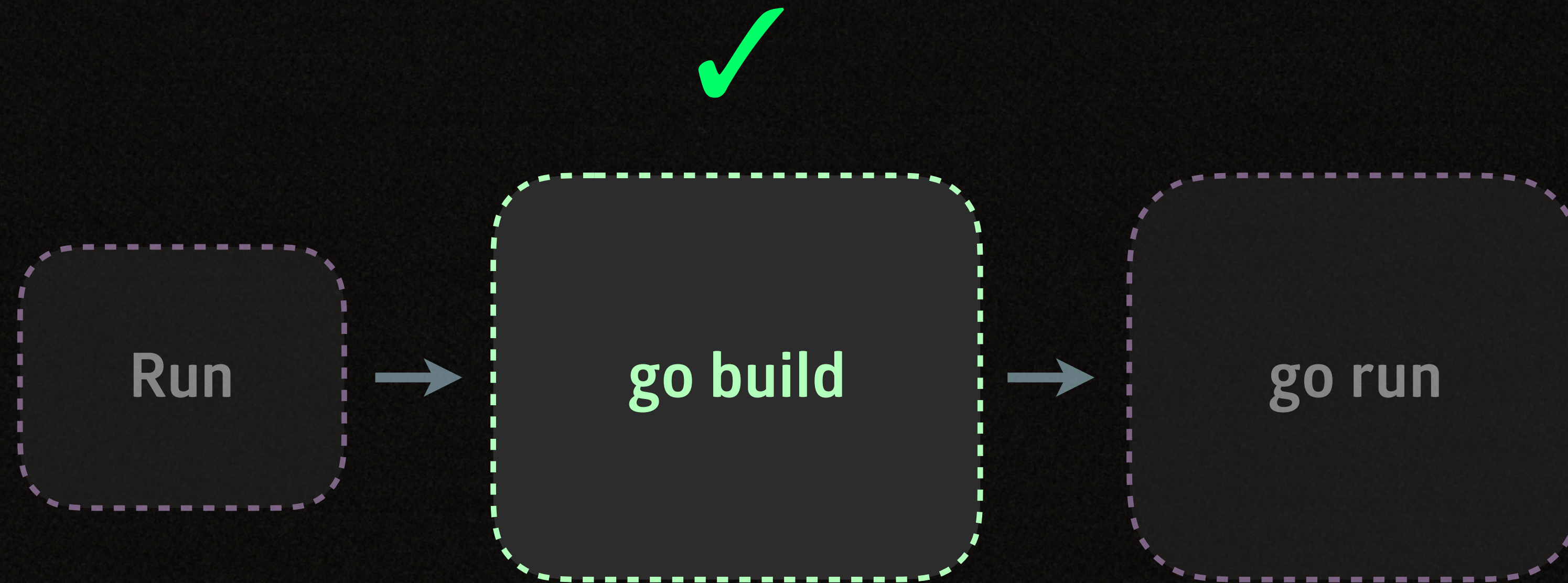
hello!

cpu



GO BUILD

You've learned how to *compile* and *run* your first Go program using ``go build`` tool.



GO RUN

You're going to learn how to *compile* your Go program using `go run` tool.



`go run main.go`



**DEMO
TIME!**

```
$ go run main.go
```

Hello Gopher!

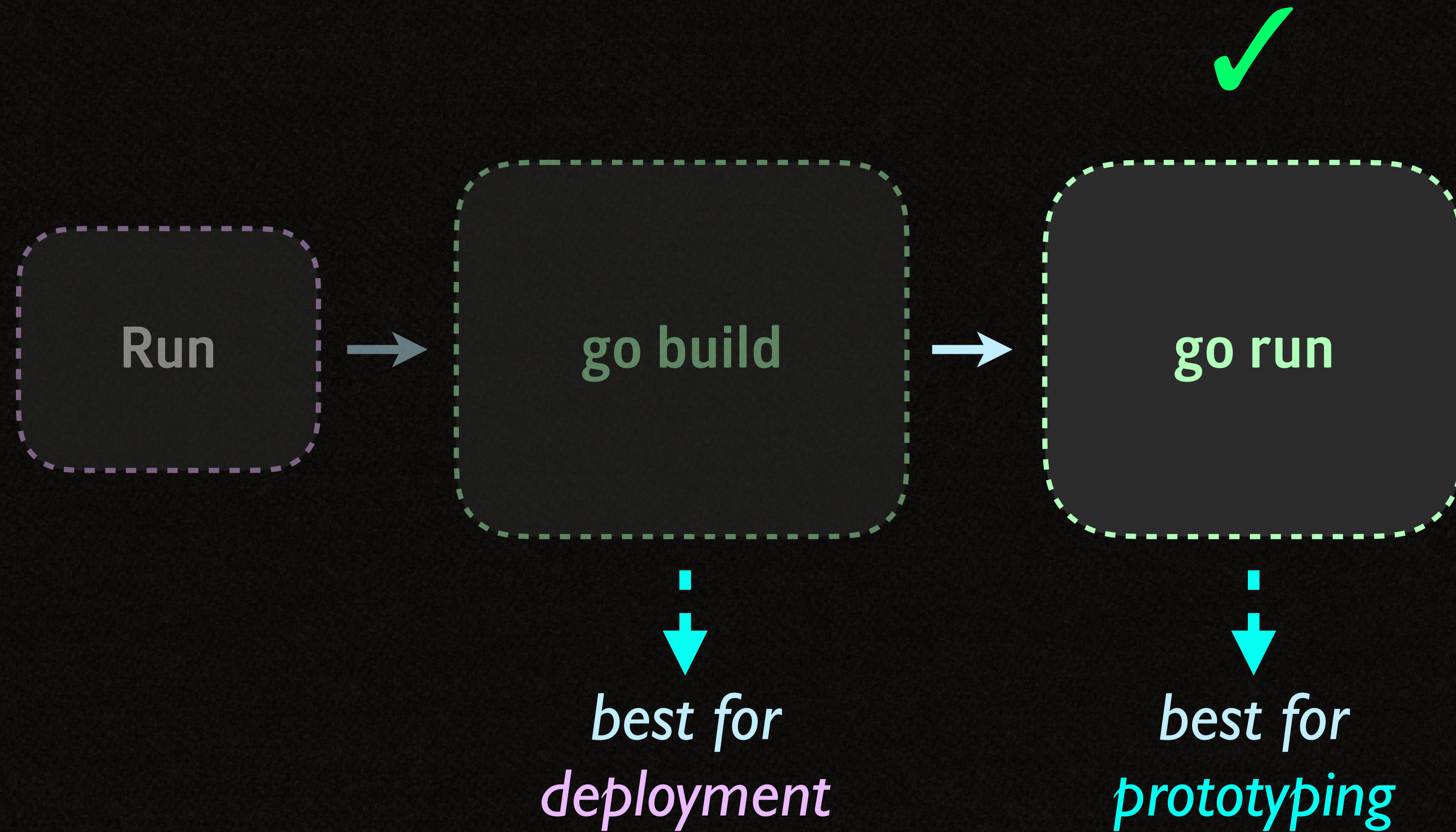
I love animations

And colors...

> Please embrace this

GO RUN

You're going to learn how to *compile* your Go program using ``go run`` tool.



FIRST PROGRAM

Congrats! You've learned how to *compile* and *run* your *first program*! That's a *great achievement*!

